UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P O Box 1450
Alexandria, Virginia 22313-1450
www uspto gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/599,402	09/28/2006	Alon Cohen	1582/8	6585

44696 7590 01/27/2011
DR. MARK M. FRIEDMAN
C/O BILL POLKINGHORN - DISCOVERY DISPATCH
9003 FLORIN WAY
UPPER MARLBORO, MD 20772

EXAMINER	
WILLIS, JONATHAN U	

ART UNIT	PAPER NUMBER
2445	

NOTIFICATION DATE	DELIVERY MODE
01/27/2011	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

mark_f@friedpat.com
nomi_m@friedpat.com
friedpat.uspto@gmail.com

Office Action Summary	**Application No.** 10/599,402	**Applicant(s)** COHEN, ALON	
	Examiner JONATHAN WILLIS	**Art Unit** 2445	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 October 2010*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *21-48* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *21-48* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *28 September 2006* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date _____.

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is responsive to the Amendments filed on 10/14/2010. Claims 21, 27, 28, 35, 42 have been amended. Claims 21-48 are pending examination.

Claim Rejections - 35 USC § 103

2. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

3. Claims 21-22, 26-29, and 33-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over US 2001/0042171 A1 to Vermeulen in view of US 2003/0195932 A1 to Tanabe et al. (hereinafter referred to as Tanabe).

4. In regard claim 21, **Vermeulen** teaches a server-side data-processing machine (*see "Remote Server,"* **in Fig. 1 [14]**) for securely and efficiently fulfilling network requests, the server-side data-processing machine comprising:

(a) a data-access engine, residing in a server memory (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) of server-side data-processing machine (*see "Remote Server,"* **in Fig. 1 [14]**), for communicating (*see* *"communication between Proxy and Remote Server,* **in Fig. 2 [12] [14] [22] [24]**) with at least one pseudo server (*see control program in Proxy Server, e.g. "proxy server 12…executes a control program stored in main memory 34…to perform the functions described in…FIG. 2,"* **in [0026] Lines 1-11**) residing in a secondary memory *(see "RAM of Proxy server,* **in Fig. 3 [34]**) of a secondary data-processing machine (*see "Proxy Server,"* **in Fig. 1 [12]**),

wherein said at least one pseudo server (*e.g. "a control program,"* **in [0026] Lines 1-11**) includes a server-logic module for fulfilling data requests (*see "File Request" and "File Transfer,"* **in Fig. 2 [21] [27]**, *e.g. "a control program…containing a sequence of control instructions to perform the functions described in…FIG. 2,"* **in [0026] Lines 1-11**) originating from a client memory (*see "requests inherently originated from the clients RAM,"* **in Fig. 4 [43]**) of a client-side data-processing machine (*see "Client,"* **in Fig. 2 [11]**),

wherein the data request from said client-side data-processing machine for data stored in said data-access engine must be routed through one of said at least one pseudo server (*see "Client" "Proxy" and "Remote Server,"* **in Fig. 1 [11] [12] [14]**, *e.g. "If client 11 wants to load a file from server 14, this request is handled via proxy server 12,"* **in [0022] Lines 9-11**),

wherein the functionality of said data access engine (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) related to said data request from said client-side data-processing machine (*see "File Request"* **in Fig. 2 [21]**) is confined to data storage and retrieval (*see functionality of the Remote Server's storage engine to store files and have files retrieved from the storage engine, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8** and *e.g. "Client 11 therefore sends a file request 21 with the address of the requested file to proxy server 12…proxy server 12 will send a "send file" request, 26, to remote server 14, which then transfers the file, 27, to the proxy server,"* **in [0024] Lines 5-17**), but

Vermeulen does not explicitly teach

wherein said at least one pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine as claimed.

However, **Tanabe** teaches

using a proxy server (*see "Proxy Server,"* **in Fig 2 [10]**) to execute an application (*see "Application,"* **in Fig. 2 [3]**) for use by a client (*see "SBC Client,"* **in Fig. 2 [6]**) in order to provide a user interface for requested content contained at a remote server (*see "SBC Remote Server,"* **in Fig. 2 [3]**, *e.g. "The SBC browser 5 displays the execution screen (i.e., the SBC browser screen), which has been produced by the SBC server 3 as the result of the execution of the application 1, and copied and distributed by the proxy server 10,"* **in [0049] Lines 4-8**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the feature of a client accessing a remote server by using a server based user interface run by a proxy server, as disclosed in **Tanabe**, into the teachings of **Vermeulen**, since both references are directed toward using a proxy server to access a remote server, hence, would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because it is well known in the art that thin-client computers are commonly used to access the internet using proxy servers to intercept data and manipulate the data in order to make remote data compatible for thick as well as thin clients (*e.g. "In recent years, a so-called SBC (server-based computing) system that allows a user to solely use an arbitrary application operated on the remote SBC server connected to the network through the SBC browser of the SBC client terminal has been developed,"* **from Tanabe in [0004]**), and **Tanabe** discloses the benefit of using server based computing to allow thin client's to execute high performance data (*e.g. "Since the client terminal simply receives the execution screen in response to the input manipulation, a low-performance terminal, such as a PDA, can be used to execute an intricate high-grade application,"* **from Tanabe**); and **Vermeulen** is enhanced by allowing different type of devices to access high performance types of remote content; thereby increasing the compatibility of the system (*e.g. "Any type of terminal device with the existing SBC browser, such as desktop PCs, note-type personal computers, PDAs, cellular phones, etc. can be used as the SBC client,"* **from Tanabe in [0105]**).

5. In regard to claim 22, **Vermeulen-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21,

wherein said data-access engine is located in a first network (*see inherent Control Program of Remote Server in a Remote Network,* **from Vermeulen in Fig. 1 [13][14]**) and at least one of said at least one pseudo one server (*see Control Program of Proxy Server, e.g. "a control program stored in main memory 34,"* **from Vermeulen in [0026] Lines 1-11**) is located in a second network (*see Network between client and Proxy*, **in Fig. 1 [11] [12]**, *e.g. "A client 11 is connected to a proxy server 12,"* **from Vermeulen in [0022] Lines 2-3**) having said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

6. In regard to claim 26, **Vermeulen-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, wherein a local data request from said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**) for data stored in one of said at least one pseudo server can be fulfilled directly by said one of said at least one pseudo server (e.*g. "If client 11 requests a file that has already been loaded and therefore is still contained in the cache, the proxy server will send this file directly from the cache to the client,"* **from Vermeulen in [0022] Lines 13-15**).

7. In regard to claim 27, **Vermeulen-Tanabe** teaches the server-side data-processing machine of claim 21, wherein said server-logic module (*e.g. "a control program...containing a sequence of control instructions to perform the functions described in...FIG. 2,"* **from Vermeulen in [0026] Lines 1-11**) and said user interface

of each of said at least one pseudo servers (*see interface of proxy to receive client output initiated by client user,* **in Fig. 3 [31],** *e.g. "Proxy server 12 has a first interface 31, which is connected to the client,"* **from Vermeulen in [0026] Lines 2-3**) further are for fulfilling logic requests (*see inherent logic contained in "File Request",* **from Vermeulen in Fig. 2 [21]**) and user interface requests (*see requests sent using user interface, e.g. "an Internet browser, for example. According to inputs by a user of the client, the control program causes files to be loaded from the distributed file system over the network into main memory,"* **from Vermeulen in [0028] Lines 3-6**) originating from said client memory (*see "requests inherently originated from the clients RAM,"* **from Vermeulen in Fig. 4 [43]**) of said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

8. Claims 28-29 and 33-34 are corresponding system claims of apparatus claims 21-22 and 26-27 respectively; therefore, they are rejected under the same rationale.

9. Claims 23-24 and 30-31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Tanabe in view of US 6,604,143 B1 to Nagar et al. (hereinafter referred to as Nagar).

10. In regard to claim 23, **Vermeulen-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 22, wherein said data-access engine is configured to communicate (*e.g. "control program adapted to…return the computed hash code to the client via an interface of server 14,"*

from Vermeulen in [0025] Lines 4-7) with other client-side data-processing machines (*e.g. "several clients are connected to such a proxy server via an internal corporate network (intranet),"* **from Vermeulen in [0023] Lines 4-5**), but

Vermeulen-Tanabe does not teach that the data-access engine communicates with other client-side data-processing machines via pseudo servers residing within said first network as claimed.

However, **Nagar** teaches the data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) communicates with other client-side data-processing machines (*see multiple computers and intranet as the client-side,* **in Fig. 2 [202] [206] [208]**) via pseudo servers residing within said first network (*see multiple proxy servers inside intranet*, **in Fig. 2 [202] [228] [230]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features of using a plurality of proxy servers to communicate with clients inside of an intranet network, as disclosed in **Nagar**, into the teachings of **Vermeulen-Tanabe**, since both of the references are directed toward proxies, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because it is well known that in distributed computer network systems (e.*g. " Computer network 13 with its servers 14 represents a distributed file system,"* **from Vermeulen in [0022]**), multiple system components may be used in a larger system to facilitate larger amount's of requested information in order to reduce the load on a single system component.

11. In regard to claim 24, **Vermeulen-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Tanabe does not teach wherein said data-access engine is configured to communicate via a content-filtering device deployed between said data access engine and said at least one pseudo server as claimed.

However, **Nagar** teaches a data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) is configured to communicate via a content-filtering device deployed between (*see "Response Filter,"* **in Fig. 2 [232] and Fig. 3 [320]**) said data access engine (*see "Server Program,"* **in Fig. 2 [220]**) and said at least one pseudo server (*see "Proxy Server,"* **in Fig. 2 [228]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features filtering content between a proxy server and a remote server, as disclosed in **Nagar**, into the teachings of **Vermeulen-Tanabe**, since both of the references are directed toward proxy caches, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because **Nagar** discloses the current problem with state of the art filtering methods (e.*g. "Generally, once the software developer writes a system that performs any filtering of information, what is filtered or how it is filtered cannot be modified except by having the software developer create a whole new filtering system,"* **from Nagar in Col. 1, Lines 41-44**) and discloses the need for improvement in content filtering (*e.g. "Therefore it is desirable to improve the filtering of information."* **from Nagar in Col. 2, Lines 4-5**), and the incorporation of **Nagar** into **Vermeulen-Tanabe** would enhance **Vermeulen-Tanabe** by allowing for current modification of filter rules in filtering incoming and outgoing proxy request data as it is well known that proxies are filtered (*e.g. "The proxy server with plug-in filters allows for easy modification of what information to filter and how to filter it,"* **from Nagar in Col. 2, Lines 11-13**).

12. Claims 30-31 are corresponding system claims of apparatus claims 23-24; therefore, they are rejected under the same rationale.

13. Claims 25 and 32 are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Tanabe in view of US 6,356,941 B1 to Cohen.

14. In regard to claim 25, **Vermeulen-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Tanabe does not teach wherein said data-access engine is configured to only fulfill said data request according to restrictions set by a network vault as claimed.

However, **Cohen** teaches a data-access engine (*see "Server's Software Module,"* **in Fig. 3 [48]**) is configured to only fulfill said data request (*e.g. "request by a* *transaction to access stored information,"* **in Col. 13, Lines 42-43**) according to restrictions set by a network vault (*e.g. Security software module 48 examines each such request to determine…whether the user has permission to perform the transaction to the particular network vault,"* **in Col. 13, Lines 43-47**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to combine the feature of restricting the retrieval of the requested stored data by using "network vaults" as disclosed in **Cohen**, into the teachings of **Vermeulen-Tanabe,** since all of the references are directed toward a accessing stored data, hence would be considered to be analogous based on their related fields of endeavor.

One would have been motivated to do so as **Cohen** discloses the problems associated with proxy servers and filtered communication and discussed the advantages of using network vaults to increase security (*e.g. "firewalls and proxy servers, can only provide filtering of communication and therefore are not sufficiently robust and secure to permit a direct connection to, and packet exchange with, limited access network 18. Therefore, if a risk is overlooked, the filter will fail. Also, the security of the firewall and/or proxy server itself can be breached, enabling the intruder to change the declarations for filtering in order to permit unauthorized access through the firewall and/or proxy server. However, the present invention does not require such packet exchange across networks, so no such declarations are needed,"* **from Cohen in Col. 7, Lines 39-50**).

15. Claim 32 is a corresponding system claim of apparatus claim 25; therefore, it is rejected under the same rationale

16. Claims 35-36 and 40-41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen in view of US 2002/0099957 A1 Kramer et al. (hereinafter referred to as Kramer) in view of Tanabe.

17. In regard to claim 35, **Vermeulen** teaches a method for securely and efficiently fulfilling network requests, the method comprising the steps of:

(a) installing a data-access engine in a server (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) of server-side data-processing machine (*see "Remote Server,"* **in Fig. 1 [14]**), for communicating (*see "communication between Proxy and Remote Server,* **in Fig. 2 [12] [14] [22] [24]**) with at least one pseudo server (*see control program in Proxy Server, e.g. "proxy server 12…executes a control program stored in main memory 34…to perform the functions described in…FIG. 2,"* **in [0026] Lines 1-11**) residing in a secondary memory *(see "RAM of Proxy server,* **in Fig. 3 [34]**) of a secondary data-processing machine (*see "Proxy Server,"* **in Fig. 1 [12]**),

wherein said at least one pseudo server (*e.g. "a control program,"* **in [0026] Lines 1-11**) includes a server-logic module (*e.g. "a control program…containing a sequence of control instructions to perform the functions* *described in…FIG. 2,"* **in [0026] Lines 1-11**) for fulfilling data requests (*see "File Request" and "File Transfer,"* **in Fig. 2 [21] [27]**) from said client-side data-processing machine for data stored in said data-access engine originating from a client memory (*see "requests inherently originated from the clients RAM,"* **in Fig. 4 [43]**) of a client-side data-processing machine (*see "Client,"* **in Fig. 2 [11]**),

wherein the functionality of said data access engine (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) related to said data request from said client-side data-processing machine (*see "File Request"* **in Fig. 2 [21]**) is confined to data storage and retrieval (*see functionality of the Remote Server's storage engine to store files and have files retrieved from the storage engine, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8** and *e.g. "Client 11 therefore sends a file request 21 with the address of the requested file to proxy server 12…proxy server 12 will send a "send file" request, 26, to remote server 14, which then transfers the file, 27, to the proxy server,"* **in [0024] Lines 5-17**), but

Vermeulen does not teach

wherein said at least one pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine; and

(b) denying said data requests unless said data requests have been routed through one of said at least one pseudo server as claimed.

However, **Kramer** teaches a firewall (see “Firewall," in Fig. 3 [311]) that blocks file requests from clients to an origin server that were not routed through a proxy server (*e.g. “The firewall 311 is configured to deny any outgoing requests that do not originate from the proxy server 312....checking that the browser client is authorized to make the request, the proxy server 312 then generates a separate request to the desired external resource on behalf of the browser client. The server that contains the desired content (called an "origin server") then receives the request. From the origin server's point of view, the proxy server 312 generated the request,”* **in [0041] Lines 1-15**).

Therefore, it would have been obvious to one of ordinary skill in the art to block incoming requests intended to be routed through a proxy server that are not routed through a proxy server, as disclosed in **Kramer**, into the teachings of **Vermeulen**, since both reference are directed toward proxying requests, hence would be considered to be analogous based on their related fields of endeavor, but

Vermeulen-Kramer does not teach

wherein said at least one pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine as claimed.

However, **Tanabe** teaches

using a proxy server (*see “Proxy Server,”* **in Fig 2 [10]**) to execute an application (*see “Application,”* **in Fig. 2 [3]**) for use by a client (*see "SBC Client,"* **in Fig. 2 [6]**) in order to provide a user interface for requested content contained at a remote server (*see "SBC Remote Server,"* **in Fig. 2 [3]**, *e.g. “The SBC browser 5 displays the* *execution screen (i.e., the SBC browser screen), which has been produced by the SBC server 3 as the result of the execution of the application 1, and copied and distributed by the proxy server 10,"* **in [0049] Lines 4-8**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the feature of a client accessing a remote server by using a server based user interface run by a proxy server, as disclosed in **Tanabe**, into the teachings of **Vermeulen**, since all of the references are directed toward directed toward proxying requests, hence, would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do combine **Kramer** with **Vermeulen** so **Vermeulen's** system is set up so that all client file requests are to be routed through a proxy, and it should be obvious to one of ordinary skill in the art to recognize that if a system is set to route all requests through a proxy, any request that is not routed through the proxy should be denied, as it is well known that firewalls are commonly set in place to avoid such unsafe routing techniques, and one would be motivated to combine **Tanabe** with **Vermeulen-Kramer** because it is well known in the art that thin-client computers are commonly used to access the internet using proxy servers to intercept data and manipulate the data in order to make remote data compatible for thick as well as thin clients (*e.g. "In recent years, a so-called SBC (server-based computing) system that allows a user to solely use an arbitrary application operated on the remote SBC server connected to the network through the SBC browser of the SBC client terminal has been developed,"* **from Tanabe in [0004]**), and **Tanabe** discloses the benefit of using server based computing to allow thin client's to execute high performance data (*e.g. "Since the client terminal simply receives the execution screen in response to the input manipulation, a low-performance terminal, such as a PDA, can be used to execute an intricate high-grade application,"* **from Tanabe**); and **Vermeulen** is enhanced by allowing different type of devices to access high performance types of remote content; thereby increasing the compatibility of the system (*e.g. "Any type of terminal device with the existing SBC browser, such as desktop PCs, note-type personal computers, PDAs, cellular phones, etc. can be used as the SBC client,"* **from Tanabe in [0105]**).

18. In regard to claim 36, **Vermeulen-Kramer-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, wherein said data-access engine is located in a first network (*see inherent Control Program of Remote Server in a Remote Network,* **from Vermeulen in Fig. 1 [13][14]**) and at least one of said at least one pseudo one server (*see Control Program of Proxy Server, e.g. "a control program stored in main memory 34,"* **from Vermeulen in [0026] Lines 1-11**) is located in a second network (*see Network between client and Proxy*, **in Fig. 1 [11] [12]**, *e.g. "A client 11 is connected to a proxy server 12,"* **from Vermeulen in [0022] Lines 2-3**) having said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

19. In regard to claim 40, **Vermeulen-Kramer-Tanabe** teaches the server-side data-

processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, wherein a local data request from said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**) for data stored in one of said at least one pseudo server can be fulfilled directly by said one of said at least one pseudo server (e.*g. "If client 11 requests a file that has already been loaded and therefore is still contained in the cache, the proxy server will send this file directly from the cache to the client,"* **from Vermeulen in [0022] Lines 13-15**).

20. In regard to claim 41, **Vermeulen-Kramer-Tanabe** teaches the server-side data-processing machine of claim 21, wherein said server-logic module (*e.g. "a control program…containing a sequence of control instructions to perform the functions described in…FIG. 2,"* **from Vermeulen in [0026] Lines 1-11**) and said user interface of each of said at least one pseudo servers (*see interface of proxy to receive client output initiated by client user,* **from Vermeulen in Fig. 3 [31],** *e.g. "Proxy server 12 has a first interface 31, which is connected to the client,"* **from Vermeulen in [0026] Lines 2-3**) further are for fulfilling logic requests (*see inherent logic contained in "File Request",* **from Vermeulen in Fig. 2 [21]**) and user interface requests (*see requests sent using user interface*, *e.g. "an Internet browser, for example. According to inputs by a user of the client, the control program causes files to be loaded from the distributed file system over the network into main memory,"* **from Vermeulen in [0028] Lines 3-6**) originating from said client memory (*see "requests inherently originated from the clients*

RAM," **from Vermeulen in Fig. 4 [43]**) of said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

21. Claims 37-38 and are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Kramer-Tanabe in view of Nagar.

22. In regard to claim 37, **Vermeulen-Kramer-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 22, wherein said data-access engine is configured to communicate (*e.g. "control program adapted to…return the computed hash code to the client via an interface of server 14,"* **from Vermeulen in [0025] Lines 4-7**) with other client-side data-processing machines (*e.g. "several clients are connected to such a proxy server via an internal corporate network (intranet),"* **from Vermeulen in [0023] Lines 4-5**), but

Vermeulen-Kramer-Tanabe does not teach that the data-access engine communicates with other client-side data-processing machines via pseudo servers residing within said first network as claimed.

However, **Nagar** teaches the data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) communicates with other client-side data-processing machines (*see multiple computers and intranet as the client-side,* **in Fig. 2 [202] [206] [208]**) via pseudo servers residing within said first network (*see multiple proxy servers inside intranet*, **in Fig. 2 [202] [228] [230]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features of using a plurality of proxy servers to communicate with clients inside of an intranet network, as disclosed in **Nagar**, into the teachings of **Vermeulen-Kramer-Tanabe**, since all of the references are directed toward proxy servers, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because it is well known that in distributed computer network systems (e.*g.* *" Computer network 13 with its servers 14 represents a distributed file system,"* **from Vermeulen in [0022]**), multiple system components may be used in a larger system to facilitate larger amount's of requested information in order to reduce the load on a single system component.

23. In regard to claim 38, **Vermeulen-Kramer-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Kramer-Tanabe does not teach wherein said data-access engine is configured to communicate via a content-filtering device deployed between said data access engine and said at least one pseudo server as claimed.

However, **Nagar** teaches a data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) is configured to communicate via a content-filtering device deployed between (*see "Response Filter,"* **in Fig. 2 [232] and Fig. 3 [320]**) said data access engine (*see*

"Server Program," **in Fig. 2 [220]**) and said at least one pseudo server (*see "Proxy Server,"* **in Fig. 2 [228]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features filtering content between a proxy server and a remote server, as disclosed in **Nagar**, into the teachings of **Vermeulen-Kramer-Tanabe**, since all of the references are directed toward proxy servers, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because **Nagar** discloses the current problem with state of the art filtering methods (*e.g. "Generally, once the software developer writes a system that performs any filtering of information, what is filtered or how it is filtered cannot be modified except by having the software developer create a whole new filtering system,"* **from Nagar in Col. 1, Lines 41-44**) and discloses the need for improvement in content filtering (*e.g. "Therefore it is desirable to improve the filtering of information."* **from Nagar in Col. 2, Lines 4-5**), and the incorporation of **Nagar** into **Vermeulen-Kramer-Tanabe** could enhance **Vermeulen-Kramer-Tanabe** by allowing for current modification of filter rules in filtering incoming and outgoing proxy request data as it is well known that proxies are filtered (*e.g. "The proxy server with plug-in filters allows for easy modification of what information to filter and how to filter it,"* **from Nagar in Col. 2, Lines 11-13**).

24. Claim 39 is rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Kramer-Tanabe in view of Cohen.

25. In regard to claim 39, **Vermeulen-Kramer-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Kramer-Tanabe does not teach wherein said data-access engine is configured to only fulfill said data request according to restrictions set by a network vault as claimed.

However, **Cohen** teaches a data-access engine (*see "Server's Software Module,"* **in Fig. 3 [48]**) is configured to only fulfill said data request (*e.g. "request by a transaction to access stored information,"* **in Col. 13, Lines 42-43**) according to restrictions set by a network vault (*e.g. Security software module 48 examines each such request to determine…whether the user has permission to perform the transaction to the particular network vault,"* **in Col. 13, Lines 43-47**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to combine the feature of restricting the retrieval of the requested stored data by using "network vaults" as disclosed in **Cohen**, into the teachings of **Vermeulen-Kramer-Tanabe,** since all of the reference are directed toward a accessing stored data, hence would be considered to be analogous based on their related fields of endeavor.

One would have been motivated to do so as **Cohen** discloses the problems associated with proxy servers and filtered communication and discussed the advantages of using network vaults to increase security (*e.g. "firewalls and proxy* *servers, can only provide filtering of communication and therefore are not sufficiently robust and secure to permit a direct connection to, and packet exchange with, limited access network 18. Therefore, if a risk is overlooked, the filter will fail. Also, the security of the firewall and/or proxy server itself can be breached, enabling the intruder to change the declarations for filtering in order to permit unauthorized access through the firewall and/or proxy server. However, the present invention does not require such packet exchange across networks, so no such declarations are needed,"* **from Cohen in Col. 7, Lines 39-50**).

26. Claims 42-43 and 47-48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen in view of US 2003/0084159 A1 to Blewett, and further in view of Tanabe.

27. In regard claim 42, **Vermeulen** teaches a server-side data-processing machine (*see "Remote Server,"* **in Fig. 1 [14]**) for securely and efficiently fulfilling network requests, the server-side data-processing machine comprising:

(a) a data-access engine, residing in a server memory (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) of server-side data-processing machine (*see "Remote Server,"* **in Fig. 1 [14]**) for communicating (*see "communication between Proxy and Remote Server,* **in Fig. 2 [12] [14] [22] [24]**) with at least one pseudo server (*see control program in Proxy Server, e.g. "proxy* *server 12…executes a control program stored in main memory 34…to perform the functions described in…FIG. 2,"* **in [0026] Lines 1-11**) residing in a secondary memory *(see "RAM of Proxy server,* **in Fig. 3 [34]**) of a secondary data-processing machine different from said server-side data processing machine, (*see "Remote Server," and "Proxy Server,"* **in Fig. 1 [12] [14]**),

wherein said at least one pseudo server (*e.g. "a control program,"* **in [0026] Lines 1-11**) includes a server-logic module (*e.g. "a control program…containing a sequence of control instructions to perform the functions described in…FIG. 2,"* **in [0026] Lines 1-11**) for fulfilling data requests (*see "File Request" and "File Transfer,"* **in Fig. 2 [21] [27]**) originating from a client memory (*see "requests inherently originated from the clients RAM,"* **in Fig. 4 [43]**) of a client-side data-processing machine (*see "Client,"* **in Fig. 2 [11]**), and

wherein the data request from said client-side data-processing machine for data stored in said data-access engine must be routed through one of said at least one pseudo server (*see "Client" "Proxy" and "Remote Server,"* **in Fig. 1 [11] [12] [14]**, *e.g. "If client 11 wants to load a file from server 14, this request is handled via proxy server 12,"* **in [0022] Lines 9-11**),

wherein the functionality of said data access engine (*see inherent file storage component inside Remote Server, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8**) related to said data request from said client-side data-processing machine (*see "File Request"* **in Fig. 2 [21]**) is confined to data storage and retrieval (*see functionality of the Remote Server's* *storage engine to store files and have files retrieved from the storage engine, e.g. "Servers 14…Such servers dedicated to storing files,"* **in [0025] Lines 7-8** and *e.g. "Client 11 therefore sends a file request 21 with the address of the requested file to proxy server 12…proxy server 12 will send a "send file" request, 26, to remote server 14, which then transfers the file, 27, to the proxy server,"* **in [0024] Lines 5-17**), but

Vermeulen does not teach

wherein said at least one pseudo server fulfills data requests via a first set of at least one communication protocols originating from a client memory of a client-side data-processing machine, and

wherein the pseudo server communicates with the data access engine via a second set of at least one communication protocols, and

wherein said at least one pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine as claimed

However, **Blewett** teaches

a pseudo proxy server that operates without having to modify any communication protocols by client devices (e.*g. "One of the benefits of the present invention is that the pseudo proxy 120 may be employed without having to modify any communication protocols that may be defined for the client terminal 130,"* **in [0036] Lines 1-4**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to combine the feature of using a multi protocol proxy server to handle different client communication protocols, as disclosed in **Blewett**, into the teachings of **Vermeulen** since both reference are directed toward proxying requests, hence would be considered to be analogous based on their related fields of endeavor, but

Vermeulen-Blewett does not teach

wherein said at least one pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine as claimed.

However, **Tanabe** teaches

using a proxy server (*see "Proxy Server,"* **in Fig 2 [10]**) to execute an application (*see "Application,"* **in Fig. 2 [3]**) for use by a client (*see "SBC Client,"* **in Fig. 2 [6]**) in order to provide a user interface for requested content contained at a remote server (*see "SBC Remote Server,"* **in Fig. 2 [3]**, *e.g. "The SBC browser 5 displays the execution screen (i.e., the SBC browser screen), which has been produced by the SBC server 3 as the result of the execution of the application 1, and copied and distributed by the proxy server 10,"* **in [0049] Lines 4-8**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the feature of a client accessing a remote server by using a server based user interface run by a proxy server, as disclosed in **Tanabe**, into the teachings of **Vermeulen-Blewett**, since all of the references are directed toward directed toward proxying requests, hence, would be considered to be analogous based on their related fields of endeavor.

One would have been motivated to do so because **Vermeulen** is concerned with proxying data between multiple clients (*e.g. "Furthermore, use is frequently made of so-called proxy servers which are disposed between one or more client computers and the distributed file system. These proxy servers, too, have a large cache memory in which the last loaded files are held temporarily in the hope of another access,"* **from Vermeulen in [0004]**), and **Blewett** could enhance **Vermeulen's** proxy by allowing more compatibility between different client types (*e.g. "One of the benefits of the present invention is that the pseudo proxy 120 may be employed without having to modify any communication protocols that may be defined for the client terminal 130. The addition or removal of a pseudo proxy server 120 is transparent to the client terminal 130. In this manner, the pseudo proxy server 120 operates differently than known proxies which typically have their own predefined protocol,"* **from Blewett in [0036]**), and it should be obvious to one of ordinary skill in the art to recognize that if a proxy server is cable of servicing multiple communication protocols, the communication protocols used by two different communications protocols accessing the same proxy server will result in at least one of the incoming data utilizing a different communication protocol incoming to the proxy server than the outgoing communication protocol being proxied to the intended destination; and one would be motivated to combine **Tanabe** with **Vermeulen-Blewett** because it is well known in the art that thin-client computers are commonly used to access the internet using proxy servers to intercept data and manipulate the data in order to make remote data compatible for thick as well as thin clients (*e.g. "In recent years, a so-called SBC (server-based computing) system that allows a user to solely use an arbitrary application operated on the remote SBC server connected to the network through the SBC browser of the SBC client terminal has been developed,"* **from Tanabe in [0004]**), and **Tanabe** discloses the benefit of using server based computing to allow thin client's to execute high performance data (*e.g. "Since the client terminal simply receives the execution screen in response to the input manipulation, a low-performance terminal, such as a PDA, can be used to execute an intricate high-grade application,"* **from Tanabe**); and **Vermeulen** is enhanced by allowing different type of devices to access high performance types of remote content; thereby increasing the compatibility of the system (*e.g. "Any type of terminal device with the existing SBC browser, such as desktop PCs, note-type personal computers, PDAs, cellular phones, etc. can be used as the SBC client,"* **from Tanabe in [0105]**).

28. In regard to claim 43, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, wherein said data-access engine is located in a first network (*see inherent Control Program of Remote Server in a Remote Network,* **from Vermeulen in Fig. 1 [13][14]**) and at least one of said at least one pseudo one server (*see Control Program of Proxy Server, e.g. "a control program stored in main memory 34,"* **from Vermeulen in [0026] Lines 1-11**) is located in a second network (*see Network between client and Proxy,* **in Fig. 1 [11] [12]**, *e.g. "A client 11 is connected to a proxy server 12,"* **from Vermeulen in**

[0022] Lines 2-3) having said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

29. In regard to claim 47, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, wherein a local data request from said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**) for data stored in one of said at least one pseudo server can be fulfilled directly by said one of said at least one pseudo server (e.*g. "If client 11 requests a file that has already been loaded and therefore is still contained in the cache, the proxy server will send this file directly from the cache to the client,"* **from Vermeulen in [0022] Lines 13-15**).

30. In regard to claim 48, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine of claim 21, wherein said server-logic module (*e.g. "a control program…containing a sequence of control instructions to perform the functions described in…FIG. 2,"* **from Vermeulen in [0026] Lines 1-11**) and said user interface of each of said at least one pseudo servers (*see interface of proxy to receive client output initiated by client user,* **from Vermeulen in Fig. 3 [31],** *e.g. "Proxy server 12 has a first interface 31, which is connected to the client,"* **from Vermeulen in [0026] Lines 2-3**) further are for fulfilling logic requests (*see inherent logic contained in "File Request",* **from Vermeulen in Fig. 2 [21]**) and user interface requests (*see requests sent using user interface, e.g. "an Internet browser, for example. According to inputs by*

a user of the client, the control program causes files to be loaded from the distributed file system over the network into main memory," **from Vermeulen in [0028] Lines 3-6**) originating from said client memory (*see "requests inherently originated from the clients RAM,"* **from Vermeulen in Fig. 4 [43]**) of said client-side data-processing machine (*see "Client,"* **from Vermeulen in Fig. 2 [11]**).

31 Claims 44-45 and are rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Blewett-Tanabe in view of Nagar.

32. In regard to claim 44, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 22, wherein said data-access engine is configured to communicate (*e.g. "control program adapted to…return the computed hash code to the client via an interface of server 14,"* **from Vermeulen in [0025] Lines 4-7**) with other client-side data-processing machines (*e.g. "several clients are connected to such a proxy server via an internal corporate network (intranet),"* **from Vermeulen in [0023] Lines 4-5**), but

Vermeulen-Blewett-Tanabe does not teach that the data-access engine communicates with other client-side data-processing machines via pseudo servers residing within said first network as claimed.

However, **Nagar** teaches the data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) communicates with other client-side data-processing machines (*see multiple computers and intranet as the client-side,* **in Fig. 2 [202] [206] [208]**) via pseudo servers residing within said first network (*see multiple proxy servers inside intranet*, **in Fig. 2 [202] [228] [230]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features of using a plurality of proxy servers to communicate with clients inside of an intranet network, as disclosed in **Nagar**, into the teachings of **Vermeulen-Blewett-Tanabe**, since all of the references are directed toward proxy servers, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because it is well known that in distributed computer network systems (e.*g.* *" Computer network 13 with its servers 14 represents a distributed file system,"* **from Vermeulen in [0022]**), multiple system components may be used in a larger system to facilitate larger amount's of requested information in order to reduce the load on a single system component.

33. In regard to claim 45, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Blewett-Tanabe does not teach wherein said data-access engine is configured to communicate via a content-filtering device deployed between said data access engine and said at least one pseudo server as claimed.

However, **Nagar** teaches a data-access engine (*see "Server Program,"* **in Fig. 2 [220]**) is configured to communicate via a content-filtering device deployed between (*see "Response Filter,"* **in Fig. 2 [232] and Fig. 3 [320]**) said data access engine (*see "Server Program,"* **in Fig. 2 [220]**) and said at least one pseudo server (*see "Proxy Server,"* **in Fig. 2 [228]**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to add the features filtering content between a proxy server and a remote server, as disclosed in **Nagar**, into the teachings of **Vermeulen-Blewett-Tanabe**, since all of the references are directed toward proxy servers, hence would be considered to be analogous based on their related fields of endeavor.

One would be motivated to do so because **Nagar** discloses the current problem with state of the art filtering methods (e.*g. "Generally, once the software developer writes a system that performs any filtering of information, what is filtered or how it is filtered cannot be modified except by having the software developer create a whole new filtering system,"* **from Nagar in Col. 1, Lines 41-44**) and discloses the need for improvement in content filtering (e.*g. "Therefore it is desirable to improve the filtering of information."* **from Nagar in Col. 2, Lines 4-5**), and the incorporation of **Nagar** into **Vermeulen-Blewett-Tanabe** could enhance **Vermeulen-Blewett-Tanabe** by allowing for current modification of filter rules in filtering incoming and outgoing proxy request data as it is well known that proxies are filtered (*e.g. "The proxy server with plug-in filters allows for easy modification of what information to filter and how to filter it,"* **from Nagar in Col. 2, Lines 11-13**).

34. Claim 46 is rejected under 35 U.S.C. 103(a) as being unpatentable over Vermeulen-Blewett-Tanabe in view of Cohen.

35. In regard to claim 39, **Vermeulen-Blewett-Tanabe** teaches the server-side data-processing machine (*see "Remote Server,"* **from Vermeulen in Fig. 1 [14]**) of claim 21, but

Vermeulen-Blewett-Tanabe does not teach wherein said data-access engine is configured to only fulfill said data request according to restrictions set by a network vault as claimed.

However, **Cohen** teaches a data-access engine (*see "Server's Software Module,"* **in Fig. 3 [48]**) is configured to only fulfill said data request (*e.g. "request by a transaction to access stored information,"* **in Col. 13, Lines 42-43**) according to restrictions set by a network vault (*e.g. Security software module 48 examines each such request to determine…whether the user has permission to perform the transaction to the particular network vault,"* **in Col. 13, Lines 43-47**).

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the claimed invention to combine the feature of restricting the retrieval of the requested stored data by using "network vaults" as disclosed in **Cohen**, into the teachings of **Vermeulen-Blewett-Tanabe,** since all of the reference are directed toward a accessing stored data, hence would be considered to be analogous based on their related fields of endeavor.

One would have been motivated to do so as **Cohen** discloses the problems associated with proxy servers and filtered communication and discussed the advantages of using network vaults to increase security (*e.g. "firewalls and proxy servers, can only provide filtering of communication and therefore are not sufficiently robust and secure to permit a direct connection to, and packet exchange with, limited access network 18. Therefore, if a risk is overlooked, the filter will fail. Also, the security of the firewall and/or proxy server itself can be breached, enabling the intruder to change the declarations for filtering in order to permit unauthorized access through the firewall and/or proxy server. However, the present invention does not require such packet exchange across networks, so no such declarations are needed,"* **from Cohen in Col. 7, Lines 39-50**).

Response to Arguments

36. In the Remarks Applicant argued in substance that:

(A) Vermeulen does not teach that the pseudo server includes a user interface (UI) for fulfilling queries of requests originating from a client memory of a client-side data-processing machine. **(Pages 11-13)**

In response to Argument (A), Applicant's arguments have been considered but are moot in view of the new ground(s) of rejection, due to the amendments that differentiate between different types of requests being handled by “a server logic module” and “a user interface.”

(B) It would not have been obvious to one ordinarily skilled in the art at the time of the current application, to apply the method of **Vermeulen** for loading files to provide a feature of the current invention for increasing the security of stored data. **(Page 13)**

In response to Argument (B), the recitation of “a server-side data-processing machine for securely and efficiently fulfilling network requests” has not been given patentable weight because the recitation occurs in the preamble. A preamble is generally not accorded any patentable weight where it merely recites the purpose of a process or the intended use of a structure, and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone. See *In re Hirao*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

In this case the steps provided in the claim do not recite how the present invention securely fulfills request, because as claimed there is no other path for a request to follow other than though a pseudo server, and therefore, as claimed it is unclear how the present invention is securely fulfilling requests any more than

Vermeulen, as clearly **Vermeulen** discloses transmitting a request to a storage machine's in different physical locations that uses some type of "separation of code."

Conclusion

37. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

US 2006/0047956 A1 to Calvin

US 2005/0004955 A1 to Lee et al.

US 2003/0046586 A1 to Bheemarasetti et al.

38. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL.** See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

39. Any inquiry concerning this communication or earlier communications from the examiner should be directed to JONATHAN WILLIS whose telephone number is (571)270-7467. The examiner can normally be reached on 8:00 A.M. - 6:00 P.M..

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Andrew Caldwell can be reached on (571)272-3868. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/JONATHAN WILLIS/
Examiner, Art Unit 2445
12/15/2010

/HASSAN PHILLIPS/

Primary Examiner, Art Unit 2445